

# Notice of Allowability

Application No.

10/053,012

Examiner

Joseph E. Avellino

Applicant(s)

FREIMUTH ET AL.

Art Unit

2143

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to Amendment dated 7/1/07.
2. ☒ The allowed claim(s) is/are 1-9, 12-14, 18-20, 22, 23.
3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
  - a) ☐ All b) ☐ Some\* c) ☐ None of the:
    1. ☐ Certified copies of the priority documents have been received.
    2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
    3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

\* Certified copies not received: \_\_\_\_\_.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.

**THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.**

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.
5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
  - (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
    - 1) ☐ hereto or 2) ☐ to Paper No./Mail Date \_\_\_\_\_.
  - (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date \_\_\_\_\_.

Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).
6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

## Attachment(s)

1. ☒ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date \_\_\_\_\_
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☒ Interview Summary (PTO-413), Paper No./Mail Date herewith.
7. ☒ Examiner's Amendment/Comment
8. ☒ Examiner's Statement of Reasons for Allowance
9. ☐ Other \_\_\_\_\_.

### EXAMINER'S AMENDMENT

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Louis Herzberg on September 5, 2007.

The application has been amended as follows:

1. (Currently amended) A method comprising:

employing at least one system for differentiating at least one service class in a kernel providing service differentiation as a kernel service based on application level information, and using service differentiation to provide different levels of quality of service for system performance to users to perform service differentiation based on content in at least one data packet for connections accepted in said at least one system, the step of employing providing content aware application header-based service differentiation in a Web server which communicates with clients over a network protecting the Web server against overload by controlling the amount and rate of work entering the system, and the step of employing including the steps of:

capturing said at least one data packet until a complete application header is detected;

parsing said complete application header to determine at least one application tag within the kernel which include classification and action rules;

matching said at least one application tag to at least one matching rule;

determining a presence of at least one application tag match with said at least one matching rule;

performing service differentiation action based on said at least one matching rule in order to provide a particular level of service from said different levels of service; and

deleting and adding rules based upon a user request; and

detecting establishment of a new connection for the purpose of service differentiation based on application layer information and providing admission control and service differentiation based on connection and application level information, wherein said step of detecting includes establishing of a new TCP connection and detecting for the purpose of service differentiation based on application layer information.

2. (Currently amended) A method as in claim 1, wherein said at least one application tag includes at least one tag taken from a group of tags including: URI, cookie, request method, HTTP version, and a tag in an application protocol.

3. (previously presented) A method as in claim 1, wherein said at least one application tag is a URI, and wherein the URI is the second string in a HTTP header as defined in an application protocol.

4. (previously presented) A method as in claim 1, further comprising employing a table having said at least one matching rule based on application layer information.

5. (previously presented) A method as in claim 1, wherein the step of determining includes finding a best match of content for application layer information.

6. (previously presented) A method as in claim 1, wherein said step of performing service differentiation action includes at least one action taken from a group of actions including: rate controlling scheduling connections, monitoring, request prioritization, and a policing action.

7. (previously presented) A method as in claim 1, wherein said step of performing service differentiation action includes an action of dropping, and wherein said action of dropping includes discarding a connection based on rules that are created to provide better performance to the connections that are accepted.

8. (original) A method as in claim 6, wherein said action includes at least one act taken from a group of acts including: sending a reset message, sending an application return code, determining compliance with a given rate and/or burst, prioritization, weighted round robin, round robin, ordering, recording statistics, performing a cleanup, and protocol control.

9. (previously presented) A method as in claim 1, further comprising installing at least one matching rule to provide a higher level of system performance for higher classed packets and connections based on application layer information.

10. - 11. (Canceled)

12. (Currently amended) A method as in claim 1 ~~claim 11~~, wherein said step of establishing of a new TCP connection includes for application header based service differentiation: receiving SYN packet; sending SYN-ACK packet; deferring accept; receiving ACK for SYN-ACK packet; and deferring notification of data packet using the 3-way handshake.

13. (original) A method as in claim 1, wherein said step of capturing includes detecting application header delimiters for said at least one data packet.

14. (Currently amended) An apparatus comprising a service differentiation module employing at least one system for differentiating at least one service class in a kernel providing service differentiation as a kernel service based on application level information, and using service differentiation to provide different levels of quality of service for system performance to users for connections accepted in said at least one system, and providing content aware application

Art Unit: 2143

header-based service differentiation in a server which communicates with clients over a network protecting the server against overload by controlling the amount and rate of work entering the system, and the step of employing includes said module including a tangible computing medium enabling functions of:

provides admission control and service differentiation based on connection and application level information

a parser to parse a client Web request;

a classifier to classify the request based on application headers and assigning a request class within a kernel;

a selector to determine an action rule based on the request class;~~and~~

a performer to apply the action rule based on the request class in order to provide better system performance for higher classed packets and connections; and

a detector to detect establishment of a new connection for the purpose of service differentiation based on application layer information and providing admission control and service differentiation based on connection and application level information, wherein the detector includes means for establishing of a new TCP connection and detecting for the purpose of service differentiation based on application layer information.

15. - 17. (Canceled)

18. (original) An article of manufacture comprising a computer usable medium having computer readable program code means embodied therein for causing service differentiation, the computer readable program code means in said article of manufacture comprising computer readable program code means for causing a computer to effect the steps of claim 1.

19. (original) A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for service differentiation, said method steps comprising the steps of claim 1.

20. (previously presented) A computer program product comprising a tangible computer usable medium having computer readable program code means embodied therein for causing service differentiation, the computer readable program code means in said computer program product comprising computer readable program code means for causing a computer to effect the functions of claim 14.

21. (Canceled)

22. (Currently amended) An apparatus comprising a computing medium enabling at least one function of:

means for employing at least one system for differentiating at least one service class in a kernel to perform service differentiation based on content in at least one data packet for connections accepted in said at least one system, the means for employing providing content aware application header-based service differentiation in servers which communicate with clients over a network protecting a Web server against overload by controlling the amount and rate of work entering the system, and the means for employing comprising:

means for capturing said at least one data packet until a complete application header is detected;

means for parsing said complete application header to determine at least one application tag;

means for matching said at least one application tag to at least one matching rule;

means for determining a presence of at least one application tag match with said at least one matching rule; and

means for performing quality of service differentiation action based on said at least one matching rule; and

means for detecting establishment of a new connection for the purpose of service differentiation based on application layer information and providing admission control and service differentiation based on connection and application level information, wherein the means for detecting includes means for establishing of a new TCP connection and detecting for the purpose of service differentiation based on application layer information.

23. (previously presented) A computer program product comprising a tangible computer usable medium having computer readable program code means embodied therein for causing differentiation of at least one service class in a kernel, the computer readable program code means in said computer program product comprising computer readable program code means for causing a computer to effect the functions of claim 22.

## **REASONS FOR ALLOWANCE**

2. The following is an examiner's statement of reasons for allowance: The prior art of record does not provide for, nor suggests providing for a system to provide service differentiation entirely within the kernel of a computer based entirely on application level information, and using the service differentiation to provide different levels of quality of service for system performance to users based on content in at least one data packet. The system captures data packets until a complete application header is detected, parsing the complete application header to determine at least one application tag within the kernel, the kernel includes classification and action rules which are added and deleted based on user's requests to the system, matching tags to at least one matching rule, determining a presence of the application tag match with the rule, and performing service differentiation action based on the matching rule in order to provide a particular level of service out of a plurality of levels of service. The system detects establishment of a new connection for the purpose of service differentiation based on application layer information and providing admission control and service differentiation based on connection and application level information, the step of detecting includes establishing of a new TCP connection and detecting for the purpose of service differentiation based on application layer information. The service differentiation mechanisms are placed in the kernel's networking stack to transparently intercept the data packets of a new TCP connection to parse the application layer headers, classify the attributes in the header based on the classifier rules and find a matching rule and the apply the associated



Art Unit: 2143

service differentiation rules that include controlling the rate and burst of new incoming requests, dropping a request, scheduling request accept order, monitoring and recording statistics. The classifier parses the HTTP header in the kernel and applies an action rule which includes waking the sleeping server process only after a decision on how to service the connection is made based on the action rule. The incoming TCP connection is classified using a SYN classifier and placed in the SYN queue that includes new connections that are not yet established. After the TCP handshake completes and data is received to determine the HTTP header, the kernel classifies the request based on the header values. Such a classification matches the incoming connection values (i.e. URIs and cookies) to a set of classification rules and their associated action rules. These are stored in a rule table 211 which are populated by a user level policy agent which uses a socket or system call to communicate with the kernel. The user space policy agent 901 uses an API to communicate with the communicator 902 in the kernel. The communicator communicates the commands from the policy agent to the initialiser 903 which sets up the service differentiation rule table, and the manager, which adds new rules and deletes and updates existing rules based on the policy agent commands.

3. The closest prior art of would be Goyal, however Goyal does not suggest providing for parsing application layer headers, rather provides service differentiation based on transport header information (i.e. IP address, etc.) and not on application layer information. Vaid discusses using application layer tags, however does not utilize these tags in a service differentiation system located in the kernel of an operating system.

Art Unit: 2143

Goyal, nor Vaid, do not teach how to collect various transport data packets to receive a complete application header transparently and entirely within the kernel stack in order to utilize the application layer tags of Vaid in the kernel of Goyal. For these reasons, in conjunction with the other limitations of the independent claims, puts this case in condition for allowance.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Joseph E. Avellino whose telephone number is (571) 272-3905. The examiner can normally be reached on Monday-Friday 7:00-4:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, David A. Wiley can be reached on (571) 272-3923. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2143

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

A handwritten signature in black ink, appearing to read 'J. Avellino', is written over the printed name.

Joseph E. Avellino, Examiner  
September 6, 2007